



# Ingénierie dirigée par les modèles pendant la phase de spécification du besoin

Benjamin Chevallereau, Alain Bernard, Pierre Mévellec

## ► To cite this version:

Benjamin Chevallereau, Alain Bernard, Pierre Mévellec. Ingénierie dirigée par les modèles pendant la phase de spécification du besoin. Ingénierie Dirigée par les Modèles, Mar 2009, Nancy, France. hal-00392383

**HAL Id: hal-00392383**

**<https://hal.science/hal-00392383>**

Submitted on 8 Jun 2009

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

---

# Ingénierie dirigée par les modèles pendant la phase de spécification du besoin

**Benjamin Chevallereau<sup>\*,\*\*\*</sup> — Alain Bernard<sup>\*</sup> — Pierre Mévellec<sup>\*\*</sup>**

<sup>\*</sup> École Centrale de Nantes - IRCCyN - 1, rue de la Noë - BP 92 101 - 44321 Nantes Cédex 03

{benjamin.chevallereau,alain.bernard}@irccyn.ec-nantes.fr

<sup>\*\*</sup> IAE de Nantes - Rue de la Censive du Tertre - BP 62 232 - 44322 Nantes Cédex 3  
pierre.mevelllec@univ-nantes.fr

<sup>\*\*\*</sup> BlueXML - 40, boulevard Jean Ingres - 44100 Nantes

---

**RÉSUMÉ.** L'ingénierie dirigée par les modèles a aujourd'hui montré la majorité de ces résultats dans la phase de développement logiciel et tout particulièrement avec l'approche MDA<sup>TM</sup>. Tandis que cette phase est grandement étudiée par la communauté IDM, la phase de spécification et d'expression du besoin est, aujourd'hui, peu approfondie. Notre proposition repose sur la mise en œuvre de l'ingénierie dirigée par les modèles dans cette phase, qui semble être l'une des plus importantes dans le processus de développement logiciel, avec pour objectif d'améliorer la qualité de la spécification des besoins et ainsi apporter une information plus fiable et plus claire aux étapes suivantes. Cette proposition repose sur un méta-modèle de spécification du besoin fonctionnel et d'un mécanisme d'interprétation à l'aide de transformations de modèles.

**ABSTRACT.** Model driven engineering show, today, the most part of its results in the phase of software development and more particularly with the MDA<sup>TM</sup> approach. Whereas this phase is greatly studied by the MDE community, the phase of specification and expression of needs is, today, not many detailed by this community. Our proposition is based on the implementation of MDE to improve the quality of specification of needs and to improve the communication between technical experts and functional profiles. This proposition is based on a meta-model, a web modeler and a set of interpretations.

**MOTS-CLÉS :** IDM, Ingénierie des besoins, Modeler web, Transformation de modèles, Génération

**KEYWORDS:** MDE, Requirements engineering, Web modeler, Model transformation, Generation

---

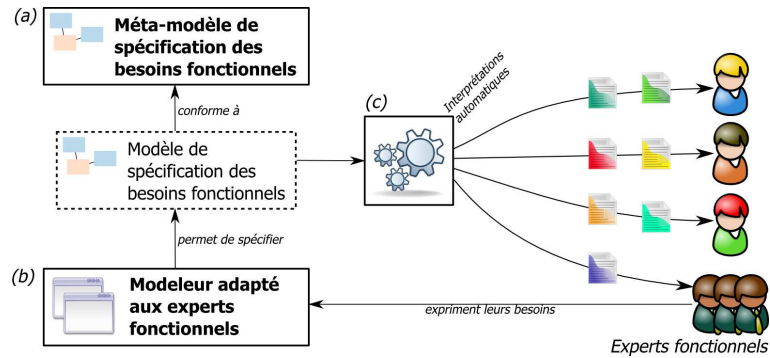
## 1. Introduction

Nous devenons de plus en plus dépendant des différents systèmes d'information auxquels nous avons accès. Cette dépendance peut nous permettre d'améliorer significativement notre efficacité mais présente également de nombreux risques de perte de productivité dans le cas d'inadéquation entre le processus et l'outil le supportant. On peut naïvement organiser l'ensemble des participants d'un processus de développement en deux catégories. La première catégorie regroupe l'ensemble des profils possédant des compétences spécifiques au métier concerné par le futur système à réaliser, c'est-à-dire les experts fonctionnels. La seconde catégorie regroupe les profils possédant des compétences techniques nécessaires à la réalisation du système, c'est-à-dire les experts techniques. Les profils appartenant aux deux catégories n'utilisent pas la même sémantique. Ils ne connaissent pas le métier de l'autre et donc ne savent pas formuler clairement leurs idées sous une forme compréhensible. L'ingénierie des besoins a donc pour objectif d'améliorer cette communication afin de mieux comprendre les besoins des experts fonctionnels et ainsi réaliser un outil complet et adapté. Pour résumer, elle a pour tâche de fournir une spécification des besoins qui doit être aussi complète et documentée que possible en utilisant des formats variés et adaptés de représentation afin d'établir une communication suffisante entre les différents participants impliqués (Pohl, 1996).

Traditionnellement, l'ingénierie des systèmes d'information se concentre sur la modélisation conceptuelle. Celle-ci vise à abstraire la spécification du système requis à partir de l'analyse des informations nécessaires à la communauté des utilisateurs (Rolland, 2007). Bien que la modélisation conceptuelle permet aux experts techniques de comprendre la sémantique du domaine, elle ne permet pas de construire des systèmes acceptés par la communauté des utilisateurs. Afin d'apporter des réponses adaptées à la communauté des utilisateurs, il est nécessaire de considérer les systèmes d'information comme un moyen d'atteindre un but déterminé dans une organisation. Comprendre ce but est une condition nécessaire à la conceptualisation d'un système. Il est important de dépasser la définition des fonctionnalités basée sur la vue du modèle conceptuel et d'étendre l'approche *que doit réaliser le système vers pourquoi le système est nécessaire*. La seconde question permet d'extraire les objectifs organisationnels et leur impact sur le système d'information.

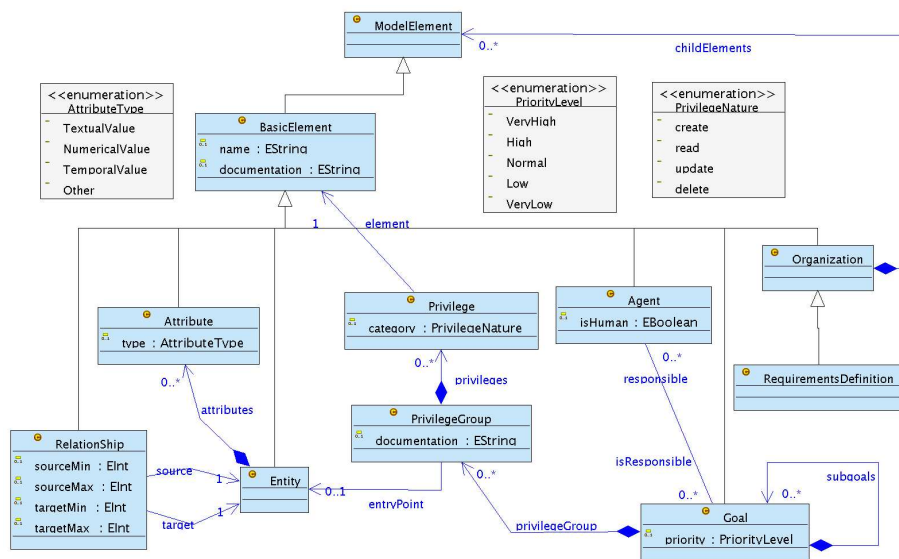
Pour résumer, notre proposition repose sur trois piliers représentés sur la figure 1. La force de cette proposition est le méta-modèle de spécification du besoin fonctionnel (a) qui est décrit plus longuement par la suite. Pour être exploité de manière optimale, celui-ci est accompagné d'un outil de modélisation (b) adapté aux experts fonctionnels. La spécification du besoin est ensuite considérée source de l'ensemble des interprétations (c). Chacune des interprétations à un public cible, une fonction donnée et un mécanisme particulier. Le concept d'interprétation sera décrit dans les sections suivantes.

La section 2 présentera le méta-modèle proposé pour la spécification d'un besoin utilisateur et le mécanisme d'interprétation sera expliqué dans la section 3.



**Figure 1.** Schéma global de notre proposition

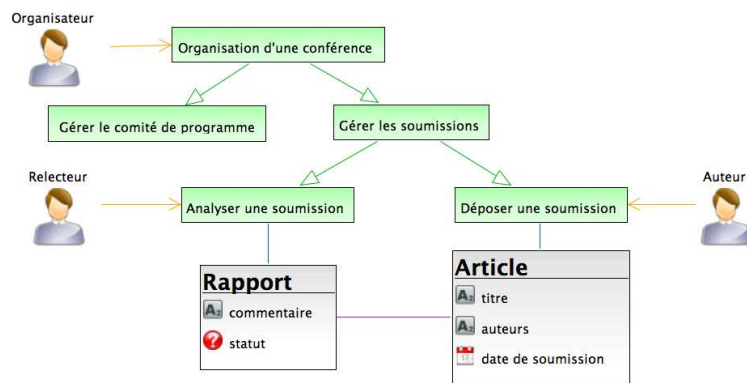
## 2. Un méta-modèle de spécification des besoins



**Figure 2.** Méta-modèle proposé pour la spécification du besoin utilisateur

Notre méta-modèle s'articule autour de sept principaux concepts : *Entity*, *Relationship*, *Attribute*, *Organization*, *Agent*, *Goal* et *Privilege*. Ils peuvent être répartis dans quatre grandes familles de modélisation : la modélisation de l'organisation qui a pour objectif de comprendre la structure dans laquelle l'application sera utilisée (*Or-*

ganization et Agent) ; la modélisation des objectifs qui permet de décrire l'activité qui sera supportée par l'outil métier (*Goal*) ; la modélisation de la structure d'information qui est utile afin de décrire les différents termes utilisés, c'est-à-dire le dictionnaire de données utilisé dans l'entreprise (*Entity, Relationship* et *Attribute*) ; la modélisation des moyens qui permet de définir les moyens à mettre en œuvre afin d'atteindre un objectif (*Privilege*).



**Figure 3.** Exemple de spécification du besoin dans le cas de l'organisation d'une conférence

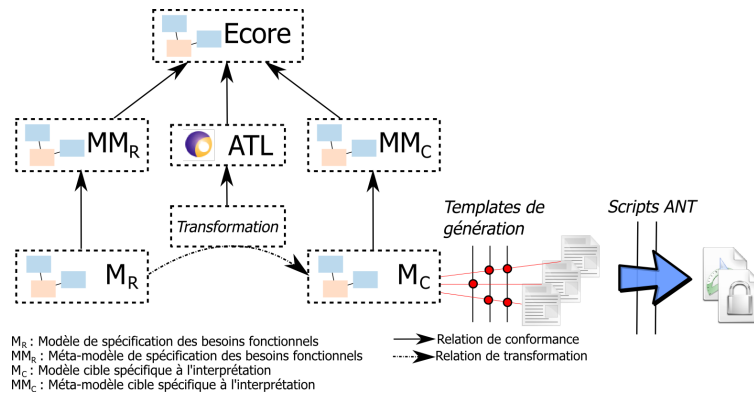
*Exemple :* On retrouve sur la figure 3 une expression partielle des besoins pour un système d'organisation de conférence. La partie à laquelle nous nous intéressons dans cet exemple est la soumission d'article. La première étape est de lire la partie concernant la modélisation des objectifs. L'objectif « Gérer les soumissions » se décompose en deux sous-objectifs. Tout d'abord, il y a la soumission avec l'objectif « Déposer une soumission » sous la responsabilité d'un auteur. Puis, il y a la re-lecture avec l'objectif « Analyser une soumission » sous la responsabilité d'un re-lecteur. La seconde étape se concentre sur la modélisation des informations, c'est-à-dire les entités et relations entre elles ainsi que les attributs caractérisant les entités. Nous pouvons considérer la définition des concepts métier comme un graphe. Celui-ci est réduit dans cet exemple à deux nœuds représentés par les entités *article* et *rapport* reliés entre eux par une relation. Après avoir défini ces deux modélisations, il est possible de les lier entre elles à l'aide des privilèges. Pour que les agents atteignent leurs buts dans le futur outil, nous allons leur définir un parcours dans le graphe des concepts, à l'aide des privilèges, pour chaque objectif sous leur responsabilité. Un parcours dans le graphe des concepts peut également être considéré comme une vue partielle sur le futur système d'information. La soumission d'un article nécessite un parcours assez simple. Il se résume à entrer dans le système par l'entité *article* puis de pouvoir en créer un nouveau. Ce parcours permet à l'auteur de renseigner le titre de l'article et ses auteurs. La re-lecture d'un article nécessite un parcours relativement plus complexe. Le point d'entrée dans le système est tout d'abord le *rapport*, un re-lecteur doit donc créer un

nouveau rapport, saisir son commentaire mais aussi définir quel est l'article qui est commenté. Ce but doit donc permettre de naviguer jusqu'au concept d'*article* pour récupérer l'information nécessaire. Malheureusement, la figure 3 ne permet pas de visualiser les privilèges accordés pour un souci de clarté du diagramme. Le seul lien visible pour le concept de privilège est le point d'entrée dans le système d'information pour chacun des objectifs.

### 3. Interprétation de l'expression du besoin

Afin d'illustrer l'intérêt du concept d'interprétation dans le contexte de spécification du besoin fonctionnel d'une application, nous pouvons utiliser le modèle de *Shannon* et *Weaver*, considéré par certains comme le modèle canonique de la communication (*Shannon*, 1948). Ce modèle est décrit par la phrase suivante : « *Un émetteur, grâce à un codage, envoie un message à un récepteur qui effectue le décodage dans un contexte perturbé de bruit.* » Si on applique ce modèle à l'industrie informatique actuelle, un expert fonctionnel, représenté comme l'émetteur, a un ensemble de besoins qu'il doit transmettre à un expert technique, représenté comme le récepteur. La spécification des besoins, représentant le message, doit donc être codée puis décodée pour ensuite être interprétée par l'expert technique. Le codage utilisé est celui des règles généralement utilisées dans le contexte de rédaction d'un cahier des charges. Celui-ci est ensuite décodé pour obtenir des spécifications techniques compréhensibles par l'expert technique. Le problème majeur de ce processus se localise dans l'étape de codage/décodage, dirigée par la maîtrise d'ouvrage, qui peut être perturbée par un bruit généré par l'environnement. Cet environnement est constitué, par exemple, des connaissances et des compétences de la maîtrise d'ouvrage mais aussi par ses expériences antérieures. Le résultat du processus de communication est donc très variable en fonction de l'étape centrale. Elle représente un facteur de risque important en raison de la quantité de bruit généré et de son impact. Afin de réduire la fragilité de ce processus, nous proposons d'automatiser un grand nombre d'interprétations. Ces interprétations vont nous permettre d'améliorer la spécification des besoins en supprimant la grande partie des bruits engendrés par une intervention humaine. Grâce à cette automatisation, nous allons également pouvoir faire intervenir un plus grand nombre de participants à l'élaboration de cette spécification. En effet, afin d'obtenir une spécification représentative des besoins réels, il est important de la faire valider par le plus grand nombre de profils dans l'entreprise. Cet ensemble peut regrouper les responsables de l'entreprise, des organisations impactées, du système d'information, de la qualité mais aussi les futurs utilisateurs.

Comme on peut le voir sur la figure 4, le mécanisme d'interprétation se décompose en trois étapes. Ce mécanisme prend en entrée un modèle de spécification des besoins fonctionnels conforme au méta-modèle proposé précédemment. La première étape est une transformation de modèles, réalisée avec ATL (*Jouault et al.*, 2006), qui permet de réaliser l'interprétation du besoin et donc concentre toute son intelligence. Cette transformation de modèles est contrainte à prendre un unique modèle en entrée conforme à



**Figure 4.** Mécanisme d'interprétation

notre méta-modèle et à retourner un unique modèle en sortie. Le méta-modèle cible est spécifique à l'interprétation. Il peut être relatif à une carte conceptuelle, à une documentation, à la modélisation UML, à l'évaluation des risques... La seconde étape est l'étape de génération. Celle-ci est réalisée à l'aide de templates de génération, en utilisant la technologie Aceleo (Obeo, 2008). Chaque template est appliqué sur le modèle issu de la transformation. Puis une étape supplémentaire est réalisée afin de réaliser des opérations difficilement réalisables par transformation ou génération à l'aide de scripts ANT (ANT, 2008). Ce mécanisme est entièrement automatisé et accessible à l'utilisateur à travers l'interface de l'outil de modélisation. Une interprétation est donc structurée sous la forme d'un triplet composé d'une transformation de modèles associée au méta-modèle cible correspondant (carte conceptuelle, documentation...), d'un ensemble de templates de génération s'appliquant sur le méta-modèle précédent puis d'un ensemble de scripts.

L'objectif de cet article n'est pas de présenter de manière approfondie l'ensemble des interprétations réalisées dans ce projet mais seulement de démontrer l'intérêt de cette approche par l'exemple. Il existe deux grandes familles d'interprétation. Il est important d'adapter la visualisation de l'expression des besoins aux différents profils qui devront valider les spécifications fonctionnelles. Cette adaptation regroupe donc l'ensemble des transformations constituant la première famille. La seconde regroupe les transformations dites de « traduction ». Leur objectif est de lire l'expression des besoins, la comprendre et enfin la traduire sous diverses formes. On peut citer les transformations d'analyse, de vérification, de documentation ou de traduction vers un formalisme technique.

#### 4. Conclusion

La première proposition du formalisme est aujourd'hui finalisée. Il permet d'exprimer son besoin pour ensuite l'interpréter correctement vers des représentations visuelles, textuelles ou techniques. Il est désormais important de le tester dans des projets industriels afin de vérifier son potentiel d'utilisabilité. Il est également important de faire valider le méta-modèle en le comparant à des formalismes existants de spécification du besoin sous forme d'objectif mais aussi à une description textuelle ou à un modèle UML. Pour cela, nous pensons utiliser les travaux de S. Patig (Patig, 2008b, Patig, 2008a) pour réaliser notre étude. Une première version de l'outil de modélisation a été réalisé en mode Web, c'est-à-dire directement utilisable dans le navigateur. Cette possibilité permet de faciliter l'expression des besoins par des experts métier en simplifiant son accès. Le temps nécessaire à sa réalisation et à sa maintenance est très important en raison de l'immaturité des technologies utilisées dans le contexte de la création d'un modèleur. De plus, la compréhension des modèles avec cette première version est très difficile en raison de l'absence de fonctionnalités indispensables. De plus, la notion de modèle et de diagramme y sont fusionnées ce qui rend la modélisation et la transformation plus complexes. Nous avons donc choisi de réaliser une nouvelle version de l'outil intégré dans l'environnement de développement Eclipse (Holzner, 2004). Celui-ci a été généré en suivant l'approche MDD/MDA proposé par le projet Topcased (Pontisso *et al.*, 2006).

#### 5. Bibliographie

- ANT, « The Apache ANT Project : <http://ant.apache.org/> », 2008.
- Holzner S., *Eclipse Cookbook*, O'Reilly Media, Inc., 2004.
- Jouault F., Kurtev I., « Transforming Models with ATL », *Satellite Events at the MoDELS 2005 Conference*, Springer Berlin / Heidelberg, p. 128-138, 2006.
- Obeo, « Acceleo Generator : <http://www.acceleo.org> », 2008.
- Patig S., « A practical guide to testing the understandability of notations », *APCCM '08 : Proceedings of the fifth on Asia-Pacific conference on conceptual modelling*, Australian Computer Society, Inc., p. 49-58, 2008a.
- Patig S., « Preparing Meta-Analysis of Metamodel Understandability », *Proceedings of the First Workshop on Empirical Studies of Model-Driven Engineering*, 2008b.
- Pohl K., *Process-Centered Requirements Engineering*, John Wiley & Sons, 1996.
- Pontisso N., Chemouil D., « TOPCASED Combining Formal Methods with Model-Driven Engineering », *ASE '06 : Proceedings of the 21st IEEE/ACM International Conference on Automated Software Engineering*, IEEE Computer Society, Washington, USA, p. 359-360, 2006.
- Rolland C., *Conceptual Modelling in Information Systems Engineering*, Springer Berlin Heidelberg, chapter Capturing System Intentionality with Maps, 2007.
- Shannon C., « A mathematical theory of communication », *Bell System Technical Journal*, 1948.



**ANNEXE POUR LE SERVICE FABRICATION**  
A FOURNIR PAR LES AUTEURS AVEC UN EXEMPLAIRE PAPIER  
DE LEUR ARTICLE ET LE COPYRIGHT SIGNÉ PAR COURRIER  
LE FICHIER PDF CORRESPONDANT SERA ENVOYÉ PAR E-MAIL

1. ARTICLE POUR LES ACTES :

*IDM 2009*

2. AUTEURS :

*Benjamin Chevallereau<sup>\*,\*\*\*</sup> — Alain Bernard<sup>\*</sup> — Pierre Mévellec<sup>\*\*</sup>*

3. TITRE DE L'ARTICLE :

*Ingénierie dirigée par les modèles pendant la phase de spécification du besoin*

4. TITRE ABRÉGÉ POUR LE HAUT DE PAGE MOINS DE 40 SIGNES :

*IDM en phase de spécification du besoin*

5. DATE DE CETTE VERSION :

*12 février 2009*

6. COORDONNÉES DES AUTEURS :

– adresse postale :

<sup>\*</sup> École Centrale de Nantes - IRCCyN - 1, rue de la Noë - BP 92 101 -  
44321 Nantes Cédex 03

{benjamin.chevallereau,alain.bernard}@irccyn.ec-nantes.fr

<sup>\*\*</sup> IAE de Nantes - Rue de la Censive du Tertre - BP 62 232 - 44322 Nantes  
Cédex 3

pierre.mevelllec@univ-nantes.fr

<sup>\*\*\*</sup> BlueXML - 40, boulevard Jean Ingres - 44100 Nantes

– téléphone : 06 22 57 51 26

– télécopie : 02 40 37 69 30

– e-mail : benjamin.chevallereau@irccyn.ec-nantes.fr

7. LOGICIEL UTILISÉ POUR LA PRÉPARATION DE CET ARTICLE :

$\text{\LaTeX}$ , avec le fichier de style `article-hermes.cls`,  
version 1.2 du 03/03/2005.

8. FORMULAIRE DE COPYRIGHT :

Retourner le formulaire de copyright signé par les auteurs, téléchargé sur :  
<http://www.revuesonline.com>

SERVICE ÉDITORIAL – HERMES-LAVOISIER  
14 rue de Provigny, F-94236 Cachan cedex  
Tél : 01-47-40-67-67  
E-mail : [revues@lavoisier.fr](mailto:revues@lavoisier.fr)  
Serveur web : <http://www.revuesonline.com>